ZILKA·KOTAB PC

ZILKA, KOTAB & FEECE™

95 SOUTH MARKET ST., SUITE 420
SAN JOSE, CA 95113

TELEPHONE (408) 971-2573
FAX (408) 971-4660

RECEIVED
CENTRAL FAX CENTER

OCT 13 2005

FAX COVER SHEET

Date:	October 13, 2005	**Phone Number**	**Fax Number**
To:	Board of Patent Appeals		(571) 273-8300
From:	Kevin J. Zilka		

Docket No.: NAI1P011/01.116.01 **App. No: 09/895,508**

Total Number of Pages Being Transmitted, Including Cover Sheet: 17

Message:

Please deliver to the Board of Patent Appeals.

Thank you,

Kevin J. Zilka

☐ *Original to follow Via Regular Mail* X *Original will Not be Sent* ☐ *Original will follow Via Overnight Courier*

IF YOU DO NOT RECEIVE ALL PAGES OR IF YOU ENCOUNTER
ANY OTHER DIFFICULTY, PLEASE PHONE Erica
AT (408) 971-2573 AT YOUR EARLIEST CONVENIENCE

October 13, 2005

RECEIVED
CENTRAL FAX CENTER

OCT 13 2005

PATENT

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re the application of	
Magdych et al.	Group Art Unit: 2136
Application No. 09/895,508	Examiner: Cervetti, David
Filed: 6/29/2001	Docket No. NAI1P011_01.116.01
For: NETWORK-BASED RISK-ASSESSMENT TOOL FOR REMOTELY DETECTING LOCAL COMPUTER VULNERABILITIES	Date: October 13, 2005

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

ATTENTION: Board of Patent Appeals and Interferences

REPLY BRIEF (37 C.F.R. § 1.193)

This Reply Brief is being filed within two (2) months of the mailing of the Examiner's Answer on September 12, 2005.

Following is an issue-by-issue reply to the Examiner's Answer.

CERTIFICATE OF MAILING/TRANSMISSION (37 C.F.R. § 1.8(a))

I hereby certify that this correspondence is, on the date shown below, being:

MAILING

☐ deposited with the United States Postal Service with sufficient postage as first class mail, in an envelope addressed to the Commissioner for Patents, Washington, D.C. 20231.

Date: 10/13/05

FACSIMILE

☒ transmitted by facsimile to the Patent and Trademark Office, (571) 273-8300.

Signature

Erica L. Farlow

(type or print name of person certifying)

Issue #1:

Group #1: Claims 1, 12 and 23-27

The Examiner first argues that it is still not clear what is configured, 1) the module or 2) the execution of the modules.

Again, appellant respectfully asserts that the claim limitations clearly state that "the commands execute the risk-assessment modules in a specific manner that is configured," (emphasis added) and thus it is the manner of execution of the modules (see 2) above) that is configured. This is clear from the claim language alone.

The Examiner continues by arguing that is not clear which device configures the modules or their execution, and how the two limitations are different. Specifically, the Examiner agues that "commands execute" and "are processed" refer to the same set of commands, but the first limitation ("the commands execute") appear to make the commands take a more active role in the execution, while the second limitation ("the commands are processed") appear to make something else, whatever extracts and executes the modules, be the active part of the execution.

It thus appears that the Examiner is arguing that the claimed "commands execute" and "are processed" are somehow inconsistent. Appellant respectfully disagrees. As claimed, in operation a) and d) of Claim 1, for example, the commands are processed on the local computer utilizing the agent (which is installed on the local computer). While the commands are claimed to execute the risk-assessment modules, they do not do so by themselves. As claimed, the commands are processed to accomplish the execution, utilizing the agent on the local computer.

In view of what is clearly claimed and the above explanation thereof, appellant asserts that the above claim limitations are indeed definite.

Issue #2:

Group #1: Claims 1, 6, 9-10, 12, 17, 20-21, 23-30, and 32

With respect to the present grouping, appellant hereby incorporates the detailed arguments of the previously filed appeal brief filed July 07, 2005 and, in the interest of a compact response to the Examiner's arguments regarding the present grouping, applicant hereby responds to the Examiner's arguments in the Examiner's Answer point-by-point below.

Regarding independent Claim 1, the Examiner now relies on the following new excerpts from Shostack to meet appellant's claimed technique "wherein the commands execute the risk-assessment modules in a specific manner that is configured at the remote computer."

> "In one aspect, the invention provides the most recent information regarding new security attacks. A user can either request the enhancement, or it can be automatically sent (e.g., via the internet) when it becomes available. The software enhancement can include a new version of the software and an update to a database of known security vulnerabilities. A user thus can obtain instant access to the latest security vulnerabilities and employ immediate remedial action before a security breach occurs." (Col. 2, lines 49-54)
>
> "The database is part of a computer security software system. The automatic update can occur whenever a software enhancement becomes available. The update can then be integrated into the computer security software. New and different security vulnerabilities are discovered almost daily. As a result, computer security checks should employ a flexible mechanism able to adapt to newly discovered security vulnerabilities. The present invention provides such a mechanism by automatically providing enhancements to a database of security vulnerabilities and using that information to provide security solutions to potentially "weak" computer networks and/or computers." (Col. 4, lines 1-12)

The Examiner continues by arguing that Shostack discloses "commands [that] execute in a manner that is configured at the remote computer." By making such argument, it appears that the Examiner still fails to take into consideration the full weight of appellant's claims. Specifically, nowhere in Shostack is there any disclosure of "commands [that] execute the risk-assessment modules in a specific manner that is configured at the remote computer" (emphasis added), as claimed, where the risk-assessment modules are claimed to be components of an agent installed on a local computer such that the risk-assessment modules are executed in a manner that is configured at a remote computer.

The Shostack excerpts above merely discuss distributing database updates including "information regarding security vulnerabilities" (see Abstract) and "new versions of software" which "require overwriting the old database of information or discarding the old version and re-installing ..." (see col. 8, lines 11-18). This in no way meets appellant's claimed locally installed agents with multiple risk-assessment modules that receive commands from a remote computer specifically for executing the risk-assessment modules in a specific manner that is configured at the remote computer.

Again, the *only* mention in Shostack of configuring anything with respect to the plurality of modules thereof, is as follows:

> "The network scan for IP devices is invoked using the properties (PROP) icon 72 which enables an authorized local user 6 to configure the various modules." (Col. 12, lines 55-57)(emphasis added)

Appellant again respectfully asserts that the above excerpt from Shostack clearly *teaches away* from appellant's claim language by disclosing a properties icon that "enables an authorized local user to configure the various modules." Appellant, on the other hand, claims that the "commands execute the risk-assessment modules in a specific manner that is configured at the remote computer."

The Examiner continues by now relying on the following new excerpts from Orchier to make a prior art showing of appellant's claimed technique "wherein the commands are processed by extracting parameters associated with the commands, and executing the risk-assessment modules indicated by the commands utilizing the associated parameters." In considering the arguments below, it should be noted that in the Final Action mailed March 01, 2005, the Examiner admits that Shostack fails to disclose the present limitations.

> "In addition to customizable queries, the query agent 82 also supports standard reports 82e, for example, accounts used after an employee is terminated and a report of users of "high risk applications". A typical standard report from the query agent 82 is shown in FIG. 8b." (Col. 13, lines 55-67)

"The manual maintenance agent 86 takes inputs from the user and converts them into platform independent security maintenance instructions which are then processed by the maintenance agent abstraction facility 90. Examples of platform independent security maintenance categories and data are as follows:
AddUserAccount(id, platformList, name, Payroll Number, expenseCode)
RemoveUserAccount(id, platformList)
AddUserAccountToGroup(id, platformList, GroupName)
RemoveUserAccountFromGroup(id, platformList, GroupName)
ModifyUserAccountName(id, platformList, name)
ModifyUserAccountPay(id, platformList, Pay)
ModifyUserAccountExpenseCode(id, platformList, expenseCode)
DisableUserAccount(id, platformList)

FIG. 8c shows the screen used to designate how often data should be collected. FIG. 8d shows the screen used to designate the server from which data should be collected. FIG. 8e shows the screen used to designate high risk applications. FIG. 8f shows the screen used to designate the environment. FIG. 8g shows the screen used to designate high risk reports. FIG. 8h shows the screen used to designate event code mapping of native codes to the common system code." (Col. 14, lines 25-52)

"28. A method of centrally controlling security in a computer network comprising a plurality of discrete computer subsystems each having a discrete security domain associated therewith, the method comprising the steps of:

separately collecting from each of the security domains security-related data associated with each security domain, wherein each security-related data is uniquely presented;

supplying the security-related data collected from the security domains to a collection agent abstraction facility and deploying the collection agent abstraction facility to transform the separately collected security-related data into a common-format security data;

storing the common-format security data in a database;

analyzing the common-format security-related data for discerning in the data out-of-compliance conditions in specific ones of said security domains by comparing the data with predetermined security regulations;

issuing common-format security-related commands effective for controlling security at the individual security domains; and

converting the common-format security-related commands to a plurality of specific security commands which are configured to be understood by corresponding ones of said security domains, the plurality of specific security commands including a specific command which results in the execution of multiple maintenance agent actions." (Claim 28)

The Examiner continues by arguing that Orchier's teachings include that the "query agent supports queries (commands with parameters)." It thus appears that the Examiner relies on Orchier's "queries" to meet appellant's claimed commands and parameters.

However, the Examiner then points to Orchier's "maintenance agent [that] takes inputs from the user and converts them to platform independent instructions." As a collateral matter, it should be noted, however, that Orchier's "queries" are used to collect data from a database and generate reports, and are not the "inputs" the Examiner notes above. As set forth in col. 14, lines 14-19 of Orchier, the user may "make changes based on manual/user inputs 87 [not queries] which are conveyed to the maintenance agent abstraction facility 90."

The Examiner continues by pointing to the following from Orchier: "the command AddUserAccount(id, platformlist, name, PayrollNumber, expensecode), the command AddUserAccount takes the parameters id, platformlist, name, PayrollNumber, expensecode."

However, the above disclosure pointed out by the Examiner merely appears to relate to allowing a user to provide input so as to make changes to a database. This clearly fails to meet appellant's claimed technique "wherein the commands are processed by extracting parameters associated with the commands, and executing the **risk-assessment modules** indicated by the commands utilizing the associated parameters."

The Examiner proceeds by pointing "to claim 28 (column 17, lines 53-63, column 18, lines 18) of the Orchier reference where "the commands are converted to a plurality of specific security commands configured to be understood by corresponding of ones of said security domains." Again, this clearly fails to meet appellant's claimed technique "wherein the commands are processed by extracting parameters associated with the commands, and executing the risk-assessment modules indicated by the commands utilizing the associated parameters."

The Examiner then concludes by exploring the possibility that Orchier does not meet the above claimed subject matter, and subsequently argues "that executing commands or modules, processing commands by extracting parameters associated with the commands was conventional

and very well known, i.e. in a Unix system, a conventional user would execute a "cd ../" command to change directories, "mkdir temp" to create a directory (folder) named "temp", or an administrator user (root) could execute a command such as "passwd" and pass as a parameter to the "passwd" command a username account to change the password for the user associated with the specified username. Computer commands in general take one or more parameters (sometimes called arguments), few commands exist that take none."

Appellant respectfully disagrees. Whether or not the foregoing blanket assertions regarding the general use of commands/parameters is true or not, Orchier fails to meet appellant's claimed technique "wherein the commands are processed by extracting parameters associated with the commands, and executing the risk-assessment modules indicated by the commands utilizing the associated parameters." Only appellant teaches and claims extracting parameters from commands for the specific purpose of **executing the risk-assessment modules** indicated by the commands utilizing the associated parameters, where the risk-assessment modules are claimed to be components of an agent installed on a local computer such that the risk-assessment modules are executed in a manner that is configured at a remote computer.

To establish a *prima facie* case of obviousness, three basic criteria must be met. First, there must be some suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art, to modify the reference or to combine reference teachings. Second, there must be a reasonable expectation of success. Finally, the prior art reference (or references when combined) must teach or suggest all the claim limitations. The teaching or suggestion to make the claimed combination and the reasonable expectation of success must both be found in the prior art and not based on appellant's disclosure. *In re Vaeck*, 947 F.2d 488, 20 USPQ2d 1438 (Fed.Cir.1991).

Appellant respectfully asserts that at least the first and third element of the *prima facie* case of obviousness has not been met, since Shostack *teaches away* from appellant's claimed invention and since the prior art references, when combined, fail to teach or suggest all of the claim limitations, as noted above.

Group #2: Claims 4 and 15

With respect to the present grouping, the Examiner relies on the following excerpt from Orchier to meet appellant's claimed "wherein the risk-assessment modules are selected for the agent based on specifications of the local computer" (see Claim 4 et al.).

> "The security domains 70a-70n communicate with collection agents 72a, 72b, 72c . . . 72n, respectively. These collection agents 72a-72n, a part of security administration system 50, represent software facilities written specifically for the corresponding operating system or system software components, for example the workstation server, LAN or NetWare.TM. software facility comprising the security domains 70a-70n. Therefore, there are many different collection agents, each of which is associated with a specific security domain type. The present invention has been reduced to practice with collection agents specific to Netware.TM. 3.1, NetWare.TM. 4.0, Windows NT, two different remote access servers, RACF, ACF2, Sybase, Oracle, AS 400, VAX/VMS, Tandem, Lotus Notes, four different UNIX operating systems and an Internet firewall.
>
> The collection agents 72a-72n use system utilities and/or APIs (Application Programming Interfaces) to extract from the individual security domains 70a-70n specific data defining security information pertaining to the system users, passwords, security groups, and where applicable: permissions, access controllers, logon events, file access events, system management events, file attributes, software and hardware versions, password control parameters, system parameters and the like. The information they collect is passed to the collection agent abstraction layer or facility 74 for further processing." (Col. 4, line 48-Col. 5, line 6 - emphasis added)

After carefully reviewing such excerpt and the remaining Orchier reference, however, it is clear that Orchier merely suggests security domains with collection agents each written for a specific domain type. Appellant notes, however, that Orchier also states that the collected data is analyzed to determine if user and system data comply with security policy requirements (Col. 7, lines 37-39). Thus, a specific collection agent is chosen according to the type of data to be collected. Choosing a collection agent for collecting data in order to further determine whether user and system data comply with security policy requirements, as taught in Orchier, simply fails to meet selecting "risk-assessment modules...based on specifications of the local computer," as claimed by appellant.

In the Examiner's Answer, the Examiner continues by arguing that Orchier teaches selecting modules based on specifications of the local computer, and in making such an argument relies on Col. 4, lines 48-62 and Col. 5, lines 1-30 thereof. The Examiner has specifically relied on Orchier's agents to meet appellant's claimed modules. The Examiner also argues that the modules in Orchier are based on the platform associated with the local computer. Appellant respectfully asserts that the modules relied on by the Examiner are simply collection agents that collect data, and therefore clearly do not meet appellant's claimed risk-assessment modules.

Again, appellant respectfully asserts that at least the third element of the *prima facie* case of obviousness has not been met, since the prior art references, when combined, fail to teach or suggest all of the claim limitations, as noted above.

Group #3: Claims 5 and 16

The Examiner relies on col. 12, lines 27-40; col. 12, lines 58-67; col. 12, lines 41-57; col. 12, lines 23-34; and col. 12, lines 14-20 from Shostack to meet appellant's claimed "wherein the risk-assessment modules include a STAT module for performing a stat system call on a file, a READ module for reading a file, a READDIR module for returning contents of a directory, a FIND module for locating a list of files based on a given function, a GETPWENT module for retrieving an entry from a password database, a GETGRENT module for retrieving an entry from a group database, a CHKSUM module for performing a checksum operation on a file, and an EXEC module for executing a command" (see Claim 5 et al.).

Appellant respectfully disagrees with the Examiner's assertions since Shostack completely fails to even suggest "a STAT module for performing a stat system call on a file, a READ module for reading a file, a READDIR module for returning contents of a directory, a FIND module for locating a list of files based on a given function...[and] a GETGRENT module for retrieving an entry from a group database."

The above cited references in Shostack merely disclose that "[t]he first module uses...[a] checksum" (col. 12, lines 27-29), "the second module performs a network scan...the network

scan produces a map of the network" (col. 12, lines 44-48), "the third module...compare[s] and identif[ies] vulnerable passwords" (Col. 12, lines 61-63). Thus, it is clear that there is no mention or suggestion in Shostack of a STAT, READ, READDIR, FIND, and/or GETGRENT module, as required by appellant's claims.

In the Examiner's Answer, the Examiner has continued to argue that appellant's modules are similar to those disclosed in Shostack since appellant's modules are utilized to detect security vulnerabilities on computers and networks. However, appellant respectfully emphasizes that what is claimed are specific modules for specific functions performed with respect to risk-assessment. Thus, whereas Shostack teaches individual modules that assess different types of network security (e.g. second module performs network scan to scan IP devices, third module assesses security vulnerabilities in passwords, etc.), appellant claims risk-assessment modules that perform specific functions within risk-assessments regardless to types of network security being assessed, namely those enumerated in the claim.

Again, appellant respectfully asserts that at least the third element of the *prima facie* case of obviousness has not been met, since the prior art references, when combined, fail to teach or suggest all of the claim limitations, as noted above.

Group #4: Claim 34

The Examiner has relied on the following excerpt from Shostack to make a prior art showing of appellant's claimed "wherein a plurality of the commands are each associated with only one of the risk-assessment modules."

> "Referring to FIGS. 5 and 6, the various integrated security system modules 160 are represented by corresponding symbols on a graphical user interface (GUI) screen 70. **The first module 74 is used to check the operating system.** The check is invoked by using the check operating systems 74' icon on the GUI screen. **The check involves ascertaining whether a user has the correct permission requirements to gain access to the network.** Also, in one embodiment of the invention, **the first module 74 determines whether all known vulnerabilities have been addressed.** Specifically, the first module 74 determines whether the suggested changes resulting from the installation procedure (Step 118)

> have been made to the operating system." (Col. 12, lines 14-26-emphasis added)

Appellant respectfully asserts that Shostack simply discloses a first module that checks an operating system, in which it is determined whether a user has correct permission to gain access to a network, and that determines whether all known vulnerabilities have been addressed (see emphasized excerpt above). Merely checking to make sure vulnerabilities have been addressed clearly does not meet appellant's claimed "plurality of the commands [that] are each associated with only one of the risk-assessment modules," since, in the above excerpt from Shostack, there is no mention of commands, in the context claimed by appellant.

In the Examiner's Answer, the Examiner has proceeded to argue that Shostack discloses various integrated security system modules. However, appellant notes that Shostack only teaches individual modules that assess different types of network security (e.g. second module performs network scan to scan IP devices, third module assesses security vulnerabilities in passwords, etc.). Shostack does not teach any commands that are "associated with only one of the risk-assessment modules," as claimed by appellant (emphasis added).

Again, appellant respectfully asserts that at least the third element of the *prima facie* case of obviousness has not been met, since the prior art references, when combined, fail to teach or suggest all of the claim limitations, as noted above.

Group #5: Claim 31

With respect to dependent Claim 31, the Examiner has relied on the following excerpt from Shostack to make prior art showing of appellant's claimed "wherein the feedback includes descriptions as to how to correct the vulnerabilities.

> "The present invention provides such a mechanism by automatically providing enhancements to a database of security vulnerabilities and using that information to provide security solutions to potentially "weak" computer networks and/or computers." (Col. 4, lines 8-12 - emphasis added)

> "The GUI 70 may also provide a reporting mechanism. The GUI 70 may also include several means for reporting various network transactions. In the disclosed invention, the GUI 70 includes a log view 80 may allow a user to view a text version the update process or log information on a storage device, a log update 82 that generates a report of all security vulnerabilities on the network 20, and a log clear function 84 that allows a user to erase the log." (col. 13, lines 36-44)

In the above cited excerpts as relied on by the Examiner, Shostack teaches automatically providing enhancements and providing security solutions to vulnerable computers. The only reporting Shostack discloses relates to logs of an update process, storage device, and security vulnerabilities. Thus, Shostack suggests automatically providing enhancements and solutions to security vulnerabilities without ever giving descriptions on how to correct the vulnerabilities. For these reasons, the Shostack reference clearly fails to meet appellant's claimed "wherein the feedback includes descriptions as to how to correct the vulnerabilities."

In the advisory action dated April 29, 2005, the Examiner responds by stating that Table 1 in Columns 5-6 shows information regarding the vulnerabilities is in the database.

Appellant respectfully asserts that the database the Examiner relies on simply includes descriptions on what vulnerabilities to check for with respect to different types of network features. For example, the database states that for a firewall, "check the firewall for vulnerability to routing, IP spoofing, and other attacks" (see specifically Col. 5, line 59). Thus, clearly the database disclosed in Shostack does not teach "feedback [that] includes descriptions as to how to correct the vulnerabilities," as claimed by appellant.

In the Examiner's Answer, the Examiner has proceeded by arguing three points. First, the Examiner argues that Col. 4, lines 8-12 in Shostack teach a database of security vulnerabilities. However, appellant claims "how to correct the vulnerabilities" and not just a database of vulnerabilities themselves. Second, the Examiner has argued that it would be obvious to one of ordinary skill in the art at the time the invention was made to generate a log including how to correct the vulnerabilities. Third, the Examiner has argued that the database of vulnerabilities as disclosed by Shostack which includes columns for "Feature" and "Vulnerability" makes it obvious to include an additional column to provide some additional information.

Appellant respectfully disagrees. Appellant specifically claims receiving feedback in response to results of a risk-assessment scan (see Claims 9 and 10 from which Claim 31 depends) where such feedback includes "descriptions as to how to correct the vulnerabilities." Thus, appellant respectfully asserts that simply because a log may be generated with found vulnerabilities, such would not make the descriptions as to how to correct the vulnerabilities obvious.

Yet again, appellant respectfully asserts that at least the third element of the *prima facie* case of obviousness has not been met, since the prior art references, when combined, fail to teach or suggest all of the claim limitations, as noted above.

Group #6: Claim 35

With respect to dependent Claim 35, the Examiner has rejected such claim limitations based on col. 4, lines 48-62 of Orchier. Appellant respectfully asserts that the Orchier reference fails to meet appellant's claimed technique "wherein a different set of risk-assessment modules exist on different local computers, based on a platform associated with each of the local computers."

In the advisory action dated April 29, 2005, the Examiner responds by stating "Orchier teaches collection agents specific to different platforms (Netware, Windows, 4 different UNIX operating systems, etc.)."

Appellant respectfully asserts that after carefully reviewing such excerpt and the remaining Orchier reference, however, it is clear that Orchier merely suggests security domains with collection agents each written for a specific domain type. Appellant notes, however, that Orchier also states that the collected data is analyzed to determine if user and system data complies with security policy requirements (Col. 7, lines 37-39). Thus, a specific collection agent is chosen according to the type of data to be collected. Choosing a collection agent for collecting data in order to further determine whether user and system data complies with security policy requirements, as taught in Orchier, simply fails to meet selecting "a different set of risk-

assessment modules [that] exist on different local computers, based on a platform associated with each of the local computers," as claimed by appellant (emphasis added).

In the Examiner's Answer, the Examiner has proceeded to argue that Orchier teaches a plurality of collection agents specific to particular platforms. The Examiner has also equated Orchier's collection agents with appellant's claimed risk-assessment modules. Appellant specifically points out Col. 4, lines 63-67 in Orchier which discloses that the "collection agents...extract...specific data defining security information pertaining to the system users, passwords, [and] security groups." Appellant respectfully asserts that collection agents that collect the type of data, as disclosed in Orchier, do not meet risk-assessment modules, as claimed by appellant.

Again, appellant respectfully asserts that at least the third element of the *prima facie* case of obviousness has not been met, since the prior art references, when combined, fail to teach or suggest all of the claim limitations, as noted above.

Issue #3

The Examiner has rejected Claims 11 and 22 under 35 U.S.C. 103(a) as being unpatentable over Shostack et al. (U.S. Patent No. 6,298,445) in view of Orchier et al. (U.S. Patent No. 6,070,244) and in further view of Smid et al.(U.S. Patent No. 4,386,233).

Group #1: Claims 11 and 22

The Examiner has relied on the following excerpt from Smid to make a prior art showing of appellant's claimed technique "wherein commands are decrypted utilizing a shared key."

> "Alternatively, authentication is accomplished by controlling access to the cryptographic function by encrypting user commands with a cryptographic function using a password supplied by the user as the cryptographic key and then **decrypting the encrypted commands using a prestored version of the password as the cryptographic key.**" (Col. 3, lines 5-12 - emphasis added)

Appellant respectfully asserts that Smid teaches "decrypting the encrypted commands using a prestored version of the password," which fails to meet the specificity of appellant's utilization of a "shared key." It is noted that a prestored version of a password as a key does not meet appellant's claimed "shared key" since a prestored version of a password does not have any bearing on whether the key is shared.

In the Examiner's Answer, the Examiner has proceeded to argue that Smid teaches using a key which is accessible only to authorized uses and authenticating a user of a cryptographic function as a condition to access an interchange key. Appellant notes that such interchange keys are "used for the exchange of keys between users" (see Col. 6, lines 26-27). Clearly, a key that is utilized to share other keys between users does not meet any sort of shared key, and especially not since appellant claims a shared key that decrypts commands.

Again, appellant respectfully asserts that at least the third element of the *prima facie* case of obviousness has not been met, since the prior art references, when combined, fail to teach or suggest all of the claim limitations, as noted above.

In view of the remarks set forth hereinabove, all of the independent claims are deemed allowable, along with any claims depending therefrom.

In the event a telephone conversation would expedite the prosecution of this application, the Examiner may reach the undersigned at (408) 971-2573. For payment of any additional fees due in connection with the filing of this paper, the Commissioner is authorized to charge such fees to Deposit Account No. 50-1351 (Order No. NAI1P011_01.116.01).

Respectfully submitted,

By: _____ Date: 10/13/05

Kevin J. Zilka
Reg. No. 41,429

Zilka-Kotab, P.C.
P.O. Box 721120
San Jose, California 95172-1120
Telephone: (408) 971-2573
Facsimile: (408) 971-4660